

I.T. Tuxtla Gutiérrez

INGENIERÍA EN SISTEMAS

ASIGNATURA

Estructura de datos

ACTIVIDAD DESARROLLADA

Síntesis y explicación de Recursividad . 2.1

ALUMNO:

NOMBRE	Nº DE CONTROL
Mario Gamaliel Vicente Domínguez	20271018

NOMBRE DEL PROFESOR:

José Amado Gómez de Paz

LUGAR: Virtual-Tuxtla

FECHA: 22/09/2021

Síntesis

La palabra “recursividad” aparece en el diccionario de la Real Academia Española haciendo referencia a la cualidad de recursivo, a lo cual la palabra “recursivo” menciona que está sujeto a reglas o pautas recurrentes.

Aunque aún no lo reconoce la Real Academia Española (Rae), de todas formas, se ha mantenido la palabra “recursividad” ya que en el lenguaje informático si se emplea de forma habitual, y aparece en cualquier referencia o manual que tenga referencia o esté relacionado con los algoritmos.

La recursividad esta en muchos sistemas del mundo real y por eso, ya que existen muchos problemas que se quiere afrontar tienen una esencia recursiva, la recursividad es una herramienta conveniente y muy útil para diseñar modelos informáticos sobre la realidad que deseamos modelizar.

Un ejemplo de esta recursividad para poder definirla sería, de la siguiente manera. Se toma un cuadrado, se divide en nueve cuadrados y se elimina el cuadro central. Con cada uno de los ocho cuadrados restantes se realiza la misma operación: dividir en nueve cuadrados y eliminar el central. Y con cada uno de los cuadrados, volver hacer lo mismo, hasta llegar hacer un bucle al infinito.

Empezando hablar en relación la programación, se indica cuando un método se llama a si mismo, en otras palabras, cuando se crea un método recursivo se debe tener en cuenta de que este tiene que llegar a terminar por lo que dentro del método debemos asegurarnos de que no se esté llamando así mismo durante su función, lo que quiere decir que el ciclo es finito.

A continuación, mencionaré las ventajas que trae el uso de la recursividad, así como también sus desventajas de manera breve y concreta.

Ventajas:

1. No es necesario definir la secuencia de pasos exactos para resolver el problema.
2. Soluciones elegantes.
3. Soluciones a problemas complejos.
4. Presenta una facilidad para comprobar y verificar que la solución es correcta.
5. La recursión agrega claridad y reduce el tiempo necesario para escribir y depurar código.
6. La recursividad es mejor en el recorrido del árbol.

Desventajas:

1. Podría ser menos eficiente.
2. Sobrecarga asociada con las llamadas a subalgoritmos.
3. El valor de la recursividad reside en el hecho de que se puede usar para resolver problemas sin fácil solución iterativa.
4. La insuficiencia inherente de algunos algoritmos recursivos.
5. Los procedimientos recursivos pueden usar mucho espacio de pila.
6. Los procedimientos recursivos mal pensados a veces pueden ser muy lentos.

Conclusión

Como conclusión, puedo decir que, la recursividad es una técnica de programación bastante útil y muy interesante de estudiar. A través de ellos el individuo, aprenderá con más rapidez y sencillez lo que es programar recursivamente e incluir esta técnica cuando se le presente un problema. La asignación de memoria, sea estática o dinámica, en realidad se tendrá que aplicar en cualquier programa al momento de su codificación; tomando en cuenta que cada programador tiene su estilo de programar.

Por otra parte, las ventajas de saber utilizarlo y aplicarlo son muchas, una de ellas es poder reducir la complejidad del tiempo, y que algunos problemas son más sencillos de modelar e implementar utilizando esta misma.

Bibliografía

A. (2011, 3 abril). *Estructura de datos*. Blog. http://adrian-estructuradedatos.blogspot.com/2011/04/memoria-estatica-y-dinamica.html

CC30A Algoritmos y Estructuras de Datos: Tipos de datos abstractos. (s. f.). Documentos Chile. Recuperado 10 de septiembre de 2021, de https://users.dcc.uchile.cl/%7Ebebustos/apuntes/cc30a/TDA/